



(12)EUROPÄISCHE PATENTANMELDUNG

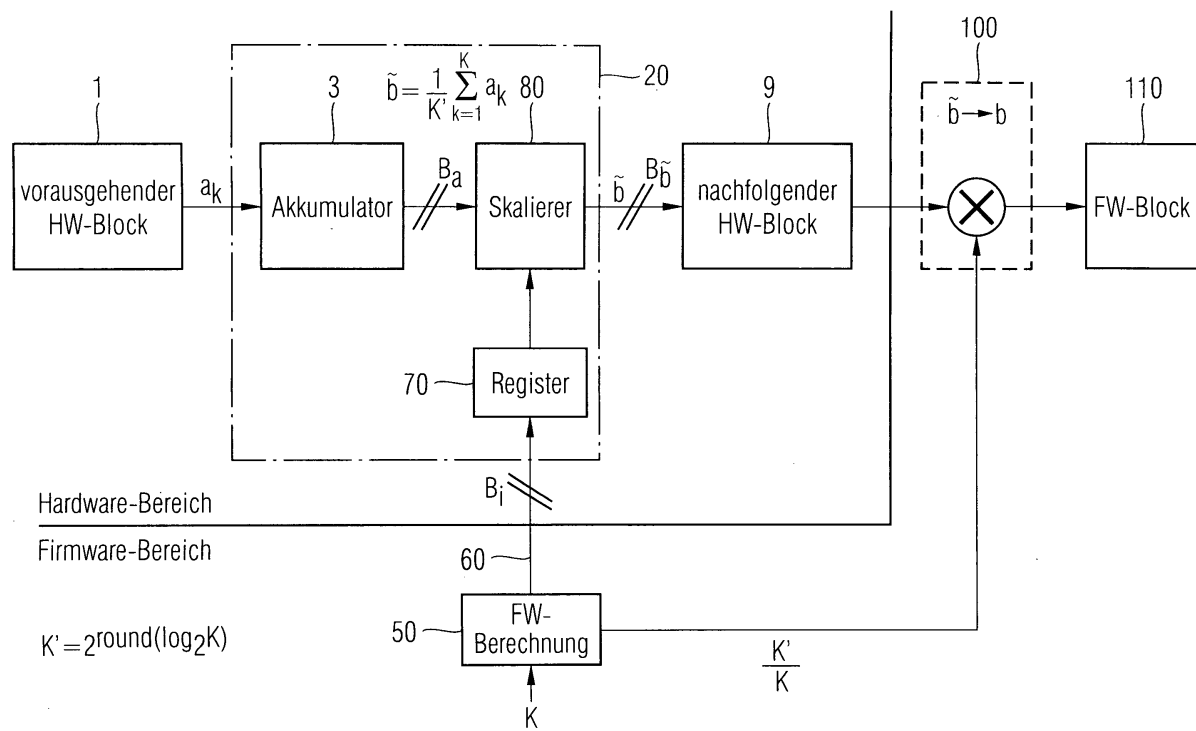
(43) Veröffentlichungstag: 21.06.2006 Patentblatt 2006/25 (21) Anmeldenummer: 05025055.4 (22) Anmeldetag: 16.11.2005	(51) Int Cl.: G06F 7/52 (2006.01)
(84) Benannte Vertragsstaaten: AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IS IT LI LT LU LV MC NL PL PT RO SE SI SK TR Benannte Erstreckungsstaaten: AL BA HR MK YU (30) Priorität: 14.12.2004 DE 102004060185 (71) Anmelder: Infineon Technologies AG 81669 München (DE)	(72) Erfinder: • Bodensterfer, Ernst 2345 Brunn am Gebirge (AT) • Drewes, Christian 82110 Germering (DE) • Niederholz, Jürgen 47647 Kerken (DE) (74) Vertreter: Patentanwälte Lambsdorff & Lange Dingolfinger Strasse 6 81673 München (DE)

(54)Verfahren und Vorrichtung zur Durchführung einer Multiplikations- oder Divisionsoperation in einer elektronischen Schaltung

(57)

Bei einem Verfahren zur Durchführung einer Multiplikations- oder Divisionsoperation $X \cdot K$ bzw. $X : K$ in einer elektronischen Schaltung wird in einem Software-Schaltungsbereich (50) der Schaltung eine Stellenverschiebung sv berechnet, derart, dass p^{sv} ein Näherungswert für K ist. Der Wert X wird in einem Hardware-Schaltungsbereich (80) bei einer Multiplikation um sv Stellen nach links bzw. bei einer Division um sv Stellen nach rechts verschoben. Im Software Schaltungsbereich (50) wird ein geeigneter Korrekturfaktor K_f berechnet. Der Wert X wird mit dem Korrekturfaktor K_f multipliziert.

FIG 2



EP 1 672 480 A1

Beschreibung

[0001] Die Erfindung betrifft ein Verfahren und eine Vorrichtung zur Durchführung einer Multiplikations- oder Divisions-Operation in einer elektronischen Schaltung. Insbesondere betrifft die Erfindung eine Schaltung zur Durchführung einer SINR-Schätzung (Signal-to-Noise and Interference Ratio) in einem Mobilfunkempfänger.

[0002] Multiplikations- oder Divisions-Operationen sind in modernen Chip-basierten Systemen (sogenannten System-on-Chip, SoC) in großer Anzahl auszuführen. Daher ist in solchen Systemen eine möglichst aufwandsgünstige Implementierung dieser Operationen von großer Bedeutung. Dabei ist zu berücksichtigen, dass die Verarbeitung von Signalen in derartigen Chip-basierten Systemen oftmals verschiedene Signalverarbeitungsphasen durchläuft und in unterschiedlichen Schaltungsbereichen durchgeführt wird. Typischerweise folgt auf eine analoge eine digitale Signalverarbeitungsphase, wobei sich die digitale Signalverarbeitung weiter unterteilen lässt in Hardware-basierte und Software-basierte Funktionalitätsrealisierung.

[0003] Analoge Signalverarbeitung wird inhärent durch analoge Schaltungen realisiert. Beispielsweise wird in einem Endgerät eines drahtlosen Kommunikationssystems das gesendete Signal über eine Antenne empfangen, im sogenannten Hochfrequenz-Frontend unter Verwendung analoger Schaltungen in das Basisband transferiert und analog/digital-umgesetzt. Die nachfolgende digitale Signalverarbeitung umfasst dann zunächst die Datendemodulation und die Decodierung, die aus Effizienzgründen oftmals weitgehend Hardware-basiert realisiert wird. Darauf folgende Signalverarbeitungsphasen werden dann zumeist auf einem digitalen Signalprozessor (DSP) oder einem Mikro-Controller, d.h. in einer Firmware- oder Software-Realisierung, programmiert, da diese Implementierungsform effizienter und/oder flexibler als die Hardware-Implementierung ist.

[0004] Im Folgenden wird lediglich der digitale Signalverarbeitungsbereich von Chip-basierten Systemen betrachtet. Eine grundlegende Entwurfsproblematik besteht darin, eine Funktionalitätsrealisierung zu finden, die die kritischen Anforderungen an den Stromverbrauch und den Chip-Flächenbedarf möglichst gut erfüllt. Hierfür ist es entscheidend, ob eine vorgegebene Funktionalität in Hardware oder Software bzw. Firmware oder verteilt realisiert wird.

[0005] Hardware-basierte Multiplikationen oder Divisionen mit reellen Zahlen treten beispielsweise bei Normierungen von in Hardware realisierten Akkumulationen von Summanden unterschiedlicher Anzahl auf. Die reellen Zahlen (Multiplikator oder Divisor) sind von der Anzahl der akkumulierten Summanden abhängig und variieren in einem a priori bekannten Zeitintervall. Beispielsweise ergibt sich im Fall von UMTS (Universal Mobile Telecommunications System) in einigen Teilen des Empfängers durch die in der UMTS Spezifikation geforderten variablen Datenraten die Schwierigkeit, dass bei Akkumulationen eine stark variierende Anzahl von Summanden auftritt, woraus die Notwendigkeit der Normierung der Akkumulationsergebnisse mit variablen Normierungsfaktoren folgt.

[0006] Fig. 1 zeigt eine konventionelle Realisierung zur Durchführung von Normierungen mit einem variablen Normierungsfaktor $1/K$. K ist hier eine ganze Zahl. Die Schaltung ist in Hardware ("Hardware-Bereich") und Firmware ("Firmware-Bereich") implementiert. Von einem vorausgehenden Hardware-Block 1 werden Datenwerte a_k eines Datenstroms bereitgestellt. k bezeichnet die diskrete Zeit. Dieser Datenstrom a_k wird einer Hardware-Schaltung 2 zur

Mittelwertbildung zugeführt. Die Hardware-Schaltung 2 führt die Berechnung des Mittelwerts $b = \frac{1}{K} \sum_{k=1}^K a_k$ durch.

Hierfür werden mittels eines Akkumulators 3 die gewünschte Anzahl von K Datenwerten a_k akkumuliert (summiert). Die

Summe $\sum_{k=1}^K a_k$ weist eine Wortbreite B_a auf. Sie wird einem Multiplizierer 4 mit der benötigten Wortbreite B_a zugeführt.

Der andere Multiplikator (Normierungsfaktor $1/K$) wird teilweise in Firmware und teilweise in Hardware berechnet. Durch Invertierung 5 in Firmware wird aus der Zahl K der Multiplikator $1/K$ ermittelt. Dieser Multiplikator $1/K$ weist eine maximale Wortbreite B_k auf. Er wird über einen Bus 6 der Wortbreite B_k an den Hardware-Bereich übergeben und dort in einem Register 7 gespeichert. Der Multiplizierer 4 greift auf das Register 7 zu und berechnet den Wert b .

[0007] Der Wert b weist eine maximale Wortbreite B_p auf. Da die Zahlen in einer Festkommadarstellung vorliegen, wird typischerweise durch einen Skalierer 8 eine nachfolgende Wortbreitenreduktion von B_p auf B_b ($B_p > B_b$) durchgeführt. Der skalierte Mittelwert b wird anschließend an einen nachfolgenden Hardware-Block 9 übergeben.

[0008] Die in Fig. 1 gezeigte Schaltung weist die folgenden Nachteile auf:

- Das Register 7 für die Speicherung des Multiplikators (hier $1/K$) und der Multiplizierer 4 müssen für die jeweiligen maximalen Wortbreiten B_k ausgeführt sein.
- Der Firmware-Hardware-Transfer des Multiplikators (hier $1/K$) über den Bus 6 muss für die maximale Wortbreite des Multiplikators und dessen maximale Änderungsfrequenz ausgelegt sein. Die Änderungsfrequenz kann für kleine Werte von K in der Größenordnung der Datenrate des Datenstroms a_k liegen, das heißt sehr groß sein.

[0009] Der Erfindung liegt die Aufgabe zugrunde, ein Verfahren zur Durchführung einer Multiplikations- oder Divisions-Operation in einer elektronischen Schaltung anzugeben, welches eine aufwandsgünstigere Durchführung dieser Operation ermöglicht. Insbesondere soll ein geringerer Stromverbrauch und eine Hardware-Ersparnis ermöglicht werden. Ferner liegt der Erfindung die Aufgabe zugrunde, eine Vorrichtung zur Durchführung einer Multiplikations- oder Divisions-Operation in einer elektronischen Schaltung, insbesondere einer Schaltung zum Durchführen einer SINR-Schätzung, mit den oben genannten Vorteilen zu schaffen.

[0010] Die der Erfindung zugrunde liegende Aufgabenstellung wird durch die Merkmale der unabhängigen Ansprüche gelöst. Vorteilhafte Ausführungsbeispiele und Weiterbildungen der Erfindung sind in den Unteransprüchen angegeben.

[0011] Bei dem erfindungsgemäßen Verfahren wird eine Multiplikations- oder Divisions-Operation $X \cdot K$ bzw. $X \cdot 1/K$ in einer elektronischen Schaltung durchgeführt, welche einen Software-Schaltungsbereich und einen Hardware-Schaltungsbereich umfasst. X und K sind reelle Zahlen in einer Zahlendarstellung zur Basis p . Bei dem erfindungsgemäßen Verfahren wird zunächst in dem Software-Schaltungsbereich aus dem Wert K eine Stellenverschiebung sv berechnet, derart, dass p^{sv} ein Näherungswert für K ist. Anschließend wird in dem Hardware-Schaltungsbereich der Wert X bei einer Multiplikation um sv Stellen nach links bzw. bei einer Division um sv Stellen nach rechts verschoben. In dem Software-Schaltungsbereich wird ein geeigneter Korrekturfaktor K_f berechnet. Der verschobene Wert von X wird mit dem Korrekturfaktor K_f multipliziert.

[0012] Dadurch, dass anstelle einer Multiplikation des Wertes X mit dem Wert K bzw. $1/K$ eine Stellenverschiebung des Wertes X durchgeführt wird, entfällt im Hardware-Schaltungsbereich die aufwändige Multiplikations-Operation. Die stattdessen auftretende Verschiebung des Wertes X kann im Hardware-Schaltungsbereich wesentlich effizienter (geringerer Flächen- und Leistungsbedarf) durchgeführt werden. Die im Vergleich zum Stand der Technik hinzukommende Multiplikation des verschobenen Wertes von X mit einem geeigneten Korrekturfaktor K_f bedeutet einen Zusatzaufwand, der jedoch im Vergleich zu der gegenüber dem Stand der Technik erreichten Ersparnis gering ist.

[0013] Die Anzahl der möglichen Werte für sv ist wesentlich geringer als die Anzahl der möglichen Werte für K . Dies bedeutet, dass der Wert sv in der Regel eine geringere Dynamik (Änderungsfrequenz) aufweist als der Wert K . Dies hat zur Folge, dass die Update-Rate der Stellenverschiebe-Operation in der Regel wesentlich geringer ist als die Update-Rate der Multiplikations-Operation im Stand der Technik. Dies führt zu einer reduzierten Buslast (geringerer Stromverbrauch) bei dem Transfer des Wertes sv vom Software-Schaltungsbereich in den Hardware-Schaltungsbereich im Vergleich zum Stand der Technik (siehe Fig. 1).

[0014] Da die Stellenverschiebung sv logarithmisch (Basis p) zu K skaliert und nur ganze Zahlen annehmen kann, ist die benötigte Wortbreite zur Darstellung der Stellenverschiebung sv (d.h. zur Steuerung der Stellenverschiebe-Operation) deutlich geringer als die Wortbreite des Wertes K bzw. $1/K$ zur Darstellung der Multiplikation im Stand der Technik (siehe Fig. 1). Dies vermindert ebenfalls die Buslast für den Transfer des Wertes sv von dem Software-Schaltungsbereich in den Hardware-Schaltungsbereich, denn es kann im Vergleich zum Stand der Technik (Fig. 1) ein Bus mit deutlich reduzierter Wortbreite eingesetzt werden.

[0015] Eine besonders vorteilhafte Ausführungsform der Erfindung kennzeichnet sich dadurch, dass der Wert $sv = \text{round}(\log_p K)$ ist, wobei $\text{round}(\cdot)$ die Rundungsoperation (Rundung des Argumentes auf die nächstliegende ganze Zahl) ist. Diese Vorschrift stellt eine besonders vorteilhafte Möglichkeit dar, eine Stellenverschiebung sv zu ermitteln. Der durch die Stellenverschiebung bestimmte Näherungswert K' für die Multiplikation beträgt in diesem Fall $K' = p^{\text{round}(\log_p K)}$.

[0016] Der Korrekturfaktor K_f sollte so gewählt werden, dass die nachfolgenden Hardware-Blöcke keine Anpassung an die erfindungsgemäße Durchführung der Multiplikations- oder Divisions-Operation erfahren müssen. Sofern ein lineares Verhalten der nachfolgenden Hardware-Blöcke vorausgesetzt werden kann, kennzeichnet sich eine vorteilhafte Ausführungsform der Erfindung dadurch, dass der Korrekturfaktor K_f bei einer Division den Wert $(p^{sv})/K$ oder bei einer Multiplikation den Wert K/p^{sv} aufweist. Zeigen die nachfolgenden Hardware-Schaltungsblöcke ein nicht-lineares Verhalten, kann der Korrekturfaktor K_f entsprechend geändert werden.

[0017] Eine weitere vorteilhafte Ausgestaltung der Erfindung kennzeichnet sich dadurch, dass der verschobene Wert X gesättigt wird, sofern dieser einen vordefinierten Zahlenwert erreicht hat. Dadurch werden Signal-Überläufe verhindert und geringstwertigste Bits (LSBs: least significant bits), die von der Genauigkeit her nicht erforderlich sind, durch Quantisieren unterdrückt.

[0018] Eine erfindungsgemäße elektronische Schaltung zur Durchführung einer Multiplikations- oder Divisions-Operation $X \cdot K$ bzw. $K \cdot 1/K$ weist ein erstes Berechnungsmittel im Software-Schaltungsbereich zum Berechnen einer Stellenverschiebung sv auf, derart, dass p^{sv} ein Näherungswert für K ist. Ferner umfasst die elektronische Schaltung einen im Hardware-Schaltungsbereich der elektronischen Schaltung enthaltenen steuerbaren Stellenverschieber (Barrel-Shifter) zum Verschieben des Wertes X bei einer Multiplikation um sv Stellen nach links bzw. bei einer Division um sv Stellen nach rechts. Der Software-Schaltungsbereich umfasst ferner zweite Berechnungsmittel zum Berechnen eines geeigneten Korrekturfaktors K_f und Mittel zum Multiplizieren des verschobenen Wertes von X mit dem Korrekturfaktor K_f .

[0019] Durch die in dieser Weise realisierte erfindungsgemäße elektronische Schaltung werden die bereits im Zusammenhang mit dem Anspruch 1 erläuterten Vorteile erreicht.

[0020] Die Erfindung wird nachfolgend anhand von Ausführungsbeispielen unter Bezugnahme auf die Zeichnungen

näher erläutert; in diesen zeigt:

Fig. 1 eine konventionelle Realisierung einer Schaltung zur Durchführung einer Divisions-Operations;

Fig. 2 ein Ausführungsbeispiel der Erfindung für die Realisierung einer erfindungsgemäßen Schaltung zur Durchführung einer Divisions-Operation; und

Fig. 3 ein Ausführungsbeispiel einer Schaltung zur Durchführung einer Signal-zu-Interferenz-und-Rausch-Schätzung, welche eine mehrfache Verwendung einer erfindungsgemäßen Schaltung vorsieht.

[0021] Fig. 2 zeigt ein Ausführungsbeispiel einer erfindungsgemäßen Schaltung zur Durchführung einer Division durch K (bzw. einer Multiplikation mit dem Faktor $1/K$). Gleiche oder ähnliche Schaltungselemente wie in Fig. 1 werden mit denselben Bezugszeichen bezeichnet. Eine dem vorausgehenden Hardware-Block 1 nachgeschaltete Hardware-Schaltung zur näherungsweise Mittelwertbildung 20 weist den Akkumulator 3, einen steuerbaren Skalierer bzw. Barrel-Shifter 80 sowie ein Register 70 auf. Der Ausgang des Akkumulators 3 weist eine Wortbreite B_a auf und ist mit dem Dateneingang des Skalierers 80 verbunden. Der Steuereingang des Skalierers 80 steht mit dem Ausgang des Registers 70 in Verbindung. Der Ausgang des Skalierers 80 verläßt die Hardware-Schaltung 20 und wird einem nachfolgenden Hardware-Block 9 zugeführt, der identisch mit dem Hardware-Block 9 in der Fig. 1 sein kann.

[0022] Der Firmware-Bereich umfasst einen Firmware-Berechnungsblock 50 zur Berechnung einer Stellenverschiebung sv. Die Stellenverschiebung sv wird über einen Bus 60 dem Hardware-Bereich übermittelt und in dem Register 70 gespeichert. Der Firmware-Berechnungsblock 50 ermittelt ferner einen Korrekturfaktor $2^{sv}/K$ in der Form $2^{\text{round}(\log_2 K)}/K$. Dieser wird einem Firmware-Korrekturblock 100 übermittelt, welcher die Ausgabedaten des Hardware-Blocks 9 prozessiert. Der auf diese Weise korrigierte Datenstrom wird in einem Firmware-Block 110 weiterverarbeitet.

[0023] Die Funktionsweise der erfindungsgemäßen Schaltung ist wie folgt:

[0024] Der einlaufende Datenstrom a_k weist eine bekannte, gegebenenfalls variable Datenrate auf. Es soll eine Akkumulation über K Datenwerte und eine entsprechende Normierung mit dem Normierungsfaktor $1/K$ durchgeführt werden, d.h. der Wert $b = \frac{1}{K} \sum_{k=1}^K a_k$ berechnet werden. Der Wert K ist variabel und kann über einen großen Wertebereich variieren.

[0025] Der im Firmware-Bereich bekannte Wert K wird (in nicht dargestellter Weise) dem Akkumulator 3 mitgeteilt und dem Firmware-Berechnungsblock 50 zur Verfügung gestellt. Der Akkumulator 3 berechnet die Summe $\sum_{k=1}^K a_k$.

Der Firmware-Berechnungs-block 50 ermittelt sv, z.B. $sv = \text{round}(\log_2 K)$, aus dem Wert K. Der Wert sv weist eine Wortbreite (Bitbreite) B_i auf und wird über einen Bus 60 in das Register 70 im Hardware-Bereich geschrieben.

[0026] Die Summe $\sum_{k=1}^K a_k$ weist die Wortbreite B_a auf. Die Stellenverschiebung sv wird von dem parametrierbaren Skalierer 80 zur Auswahl der Bits bei der auf die Summe $\sum_{k=1}^K a_k$ angewandten Schiebeoperation verwendet. Der durch die Schiebeoperation erzeugte approximierte Wert $\tilde{b} = \frac{1}{2^{sv}} \sum_{k=1}^K a_k$ weist die Wortbreite $B_{\tilde{b}}$ auf. Da $2^{sv}/K < 2$ gilt, ist die Wortbreite $B_{\tilde{b}}$ identisch mit der Wortbreite B_b des richtigen Werts b, d.h. $B_{\tilde{b}} = B_b$.

[0027] Mit anderen Worten wird die Summe $\sum_{k=1}^K a_k$ in der Hardware-Schaltung 20 nicht mit dem Korrekturfaktor $1/K$ selbst, sondern mit einem Näherungswert $1/2^{sv}$ multipliziert.

[0028] Der hierdurch eingefügte Fehler wird durch den in dem Firmware-Berechnungsblock 50 ermittelten Korrekturfaktor Kf kompensiert.

[0029] Sofern der Hardware-Block 9 ein lineares Verhalten zeigt, ist der Korrekturfaktor $Kf = 2^{sv}/K$. Für $sv = \text{round}(\log_2 K)$ ergibt sich $Kf = 2^{\text{round}(\log_2 K)}/K$. Die Kompensations-Multiplikation in dem Firmware-Korrekturblock 100 bewirkt, dass die nachfolgende Datenverarbeitung (Firmware-Block 110) mit dem exakten Wert b weiterarbeiten kann.

[0030] Die Kompensations-Multiplikation mit dem Korrekturfaktor $K_f = 2^{sv}/K$ setzt also ein lineares Verhalten des nachfolgenden Hardware-Blocks 9 voraus. Ist das Verhalten des nachfolgenden Hardware-Blocks 9 nicht linear, muss eine entsprechend angepasste Kompensations-Multiplikation erfolgen. In allen Fällen sind die im Signalweg hinter dem Firmware-Korrekturblock 100 liegenden Datenverarbeitungsblöcke (Firmware-Block 110 und/oder nicht dargestellte Hardware-Blöcke) nicht von dieser Änderung betroffen. Wie bereits erwähnt ist auch der Hardware-Block 9 identisch mit dem im Stand der Technik verwendeten Hardware-Block 9.

[0031] Die erfindungsgemäße Lösung weist die folgenden Vorteile auf:

- Die benötigte Wortbreite zur Darstellung der Schiebeoperation B_i ist üblicherweise deutlich kleiner als die Wortbreite des Faktors selbst, d.h. $B_i \ll B_K$.
- Die Update-Rate der Schiebeoperation kann kleiner ausfallen als die des Multiplikators (Normierungsfaktor $1/K$) im Stand der Technik, da der gerundete Wert $2^{\text{round}(\log_2 K)}$ von K eine geringere Dynamik aufweist als K . Dieser und der vorherige Punkt führen zu einer reduzierten Buslast am Bus 60 (verglichen mit dem Bus 6).
- Der im Stand der Technik erforderliche Multiplizierer 4 fällt weg, der Skalierer 80 wird etwas komplizierter als der Skalierer 8 im Stand der Technik. Hieraus resultiert eine deutlich weniger aufwändige Hardware.

[0032] Als ein geringfügiger Nachteil der erfindungsgemäßen Lösung kann die notwendige Firmware-Abbildung des Wertes K auf die Schiebeoperation (Stellenverschiebung sv) und den Korrekturfaktor K_f gesehen werden. Der Aufwand hierfür ist allerdings vergleichbar mit der in Fig. 1 durchzuführenden Abbildung (Invertierung) von K auf $1/K$.

[0033] Darüber hinaus ist festzustellen, dass das erfindungsgemäße Konzept besonders effizient verwendet werden kann, wenn in dem Hardware-Bereich mehrere solcher Multiplikationen oder Divisionen mit denselben Faktoren (K bzw. $1/K$) durchgeführt werden. Dann können sämtliche Operationen mit derselben Firmware-Korrekturmultiplication in dem Firmware-Korrekturblock 100 korrigiert werden. Dies wird auch aus dem folgenden Ausführungsbeispiel deutlich.

[0034] Fig. 3 zeigt in beispielhafter Weise die Anwendung der Erfindung in den Schaltungsblöcken SINR-Schätzung (Signal-to-Noise and Interference Ratio) 200 und CS-Schätzung (Channel State) 250 des MGOLD2™ Chips von der Infineon Technologies AG. Zum besseren Verständnis der für die Erfindung wesentlichen Aspekte sind die Schaltungsblöcke SINR-Schätzung 200 und CS-Schätzung 250 vereinfacht (nur für den Betriebsmodus "normal mode, databased") dargestellt.

[0035] Der Hardware-Block SINR-Schätzung 200 setzt sich aus vier Subblöcken DPCH-Leistungsberechnung 210, CPICH-Leistungsberechnung 220, CPICH-Störleistungsberechnung 230 und einer Einheit zur Berechnung des Amplitudenverhältnisses 240 zusammen. Dem Hardware-Block SINR-Schätzung 200 ist der Block 250 zur Schätzung des Kanalzustands nachgeordnet.

[0036] Die SINR-Schätzung basiert in UMTS auf den empfangenen komplexwertigen Symbolen der physikalischen Kanäle DPCH (Dedicated Physical Channel) und CPICH (Common Pilot Channel). Die Kanalstruktur von UMTS ist bekannt und z.B. in der Spezifikation 3GPP TS 25.211 V4.6.0 (2002-09) (Release 4) angegeben.

[0037] Der Ausgang des Subblocks 210 für die DPCH-Leistungsberechnung wird sowohl dem Firmware-Bereich als auch dem Subblock 240 zur Berechnung des Amplitudenverhältnisses zugeleitet. Der Ausgang des Subblocks 220 zur CPICH-Leistungsberechnung wird dem Subblock 240 zur Berechnung des Amplitudenverhältnisses zugeführt. Dessen Ausgang wird sowohl dem Firmware-Bereich als auch einem Eingang des Blocks 250 für die Kanalzustandsschätzung zugeleitet. Der Ausgang des Subblocks 230 wird dem anderen Eingang des Blocks 250 für die Kanalzustandsschätzung sowie dem Firmware-Bereich zugeführt. Der Firmware-Bereich kann beispielsweise durch einen DSP realisiert sein.

[0038] Im Folgenden werden der Aufbau und die Funktionsweise der einzelnen Subblöcke 210, 220, 230, 240 und des Blockes 250 näher erläutert:

[0039] Die Eingangssignale der Subblöcke 210, 220, 230 werden aus den entspreizten und Delay-kompensierten DPCH-Symbolen x_{data} , CPICH-Symbolen y und den CPICH-Kanalgewichten h gebildet. Die CPICH-Kanalgewichte h kann man sich als tiefpassgefilterte, entspreizte und Delay-kompensierte CPICH-Symbole y vorstellen. Sämtliche Eingangssignale sind komplexwertig und liegen für alle im UMTS-Empfänger parallel verarbeiteten Ausbreitungspfade (Rake-Finger) an, wobei durch die vorgelagerte Delay-Kompensierung die Zeitverschiebungen durch die pfadspezifischen Ausbreitungswege bereits kompensiert sind. Daher können die über verschiedene Ausbreitungspfade empfangenen Eingangssignalkomponenten als zeitlich synchron betrachtet werden.

[0040] Der Subblock 210 für die DPCH-Leistungsberechnung weist einen komplexen Quadrierer CSQR (complex square) sowie einen mehrkanaligen Akkumulator cell-accu auf. Am Ausgang des mehrkanaligen Akkumulators cell-accu ist ein steuerbarer Skalierer (Barrel-Shifter) sc12 vorgesehen. Dem steuerbaren Skalierer sc12 ist ein FIR-Filter FIR_1 nachgeschaltet.

[0041] In dem Subblock 210 zur DPCH-Leistungsberechnung wird die Signalleistung W_D gemäß der folgenden Gleichung berechnet:

$$W_D(C) = \text{FIR}_1 \left((1/K_X) \cdot \sum_{m=1 \dots M(C)} \sum_{k=1 \dots K_X} |x_{\text{data}_m}(k)|^2 \right). \quad (1)$$

[0042] Dabei bezeichnet m die Fingernummer und k den Zeitindex. C ist die Zellennummer, M(C) die Anzahl der Rake-Finger in Zelle C. $x_{\text{data}_m}(k)$ bezeichnet das k-te Datensymbol ($k = 1, \dots, K_X$) im Zeitschlitz von Rake-Finger m. K_X bezeichnet die Anzahl der Symbole eines Zeitschlitzes.

[0043] Demnach werden in dem Subblock 210 zur DPCH-Leistungsberechnung die komplexen DPCH-Datensymbole $x_{\text{data}_m}(k)$ in dem komplexen Quadrierer CSQR betragsmäßig quadriert. Die Betragsquadrate über alle Symbole eines Zeitschlitzes ($k = 1, \dots, K_X$) und alle Rake-Finger einer Zelle ($m = 1, \dots, M(C)$) werden in dem Akkumulator cell-accu aufsummiert. Für jede Zelle ergibt sich pro UMTS-Zeitschlitz ein Ergebniswert. Diese zellspezifischen Ergebniswerte werden mittels des steuerbaren Skalierers sc12 zur Normierung einer Skalierung mit dem Kehrwert der Anzahl der Symbole pro Zeitschlitz, $1/K_X$, unterzogen. K_X kann sich bei einer Änderung des Spreizfaktors an den Zeitschlitzgrenzen auch ändern. Das für jede Zelle erhaltene normierte Ergebnis wird dann in dem FIR_1 Filter tiefpassgefiltert. $W_D(C)$ gibt die DPCH-Leistung der Zelle mit der Zellennummer C an.

[0044] Der Subblock 220 zur CPICH-Leistungsberechnung ist analog zu dem Subblock 210 aufgebaut. Der Skalierer im Subblock 220 ist mit dem Bezugszeichen sc33 bezeichnet. Statt der Datensymbole $x_{\text{data}_m}(k)$ werden die Kanalgewichte $h_m(k)$ verarbeitet, wobei mit $h_m(k)$ das k-te Kanalgewicht ($k = 1, \dots, 10$) von dem Rake-Finger m bezeichnet wird. Pro Zeitschlitz gibt es konstant 10 Kanalgewichte. Daher erfolgt die Skalierung konstant mit einem Faktor $1/10$. Die Gleichung zur Berechnung der CPICH-Leistung lautet demnach

$$W_C(C) = \text{FIR}_2 \left((1/10) \cdot \sum_{m=1 \dots M(C)} \sum_{k=1 \dots 10} |h_m(k)|^2 \right). \quad (2)$$

[0045] Dabei bezeichnet $W_C(C)$ die CPICH-Signalleistung in der Zelle mit der Zellennummer C.

[0046] Der Subblock 230 für die CPICH-Rauschleistungsberechnung ist ähnlich wie die Subblöcke 210 und 220 aufgebaut. Er weist zusätzlich einen eingangsseitigen Subtrahierer sub auf. Mit $Y_m(k)$ wird das k-te CPICH-Symbol ($k = 1, \dots, 10$) von dem Rake-Finger m bezeichnet. In dem Subtrahierer sub wird die Differenz $y_m(k) - h_m(k)$ gebildet. Diese Differenz kann als geschätztes Rauschsignal aus dem CPICH aufgefasst werden, weil es durch Abziehen des tiefpassgefilterten CPICH-Signals vom originalen CPICH-Signal gebildet wird. Diese geschätzten aktuellen komplexen Rauschamplituden, $y_m(k) - h_m(k)$, werden in dem komplexen Quadrierer CSQR betragsmäßig quadriert, in dem Akkumulator cell-accu summiert, von dem Skalierer sc22 skaliert und mittels des FIR-Filters FIR_3 gefiltert. Als Rauschleistung N(C) für die Zelle mit der Zellennummer C ergibt sich

$$N(C) = \text{FIR}_3 \left((1/(10 \cdot M(C))) \cdot \sum_{m=1 \dots M(C)} \sum_{k=1 \dots 10} |y_m(k) - h_m(k)|^2 \right) \quad (3)$$

[0047] Aus obiger Gleichung wird deutlich, dass im Unterschied zu den beiden vorangegangenen Algorithmen (Gleichungen (1) und (2)) in Gleichung (3) nicht nur durch die Anzahl der CPICH-Symbole pro Zeitschlitz, d.h. durch 10 dividiert wird, sondern dass zusätzlich durch die Anzahl der Finger der Zelle, M(C), dividiert wird. Die Anzahl M(C) kann für jede Zelle verschieden sein und kann sich von Zeitschlitz zu Zeitschlitz ändern.

[0048] Alle FIR-Filter FIR_1 , FIR_2 , FIR_3 haben ein Tiefpassverhalten mit der Verstärkung 1 bei der Frequenz 0 Hz. Damit können sie für die Betrachtung der Skalierungsfaktoren außer Acht gelassen werden. Die Koeffizienten der drei FIR-Filter FIR_1 , FIR_2 , FIR_3 können verschieden sein, was durch die unterschiedlichen Indizes 1, 2, 3 angedeutet wird.

[0049] Der Subblock 240 für die Berechnung des Amplitudenverhältnisses führt eine Division der geschätzten DPCH-Signalleistung $W_D(C)$ durch die geschätzte CPICH-Leistung $W_C(C)$ und die Berechnung der Wurzel dieses Verhältnisses durch. Die so gewonnene Größe A(C) wird als Amplitudenverhältnis (Amplitude Ratio) bezeichnet. Für jeden Zeitschlitz wird für jede Zelle (Zellennummer C) ein Wert berechnet:

$$A(C) = \sqrt{W_D(C) / W_C(C)} \quad (4)$$

[0050] Der Block 250 für die Kanalzustandsschätzung umfasst eine Divisionsstufe DIV sowie einen Skalierer sce_scale. Der Dividierer DIV dividiert das Amplitudenverhältnis A(C) durch die Rauschleistung N(C). Das Ergebnis der

Division ist der Normierungsfaktor (Normalization Factor) $F(C)$, der von der in Fig. 3 nicht dargestellten MRC-Hardware-Einheit (MRC: Maximum Ratio Combiner), welche den Kombinierer des Rake-Empfängers bildet, benötigt wird. Der Normierungsfaktor $F(C)$ wird für jeden Zeitschlitz und jede Zelle nach

$$F(C) = A(C) / N(C) \quad (5)$$

berechnet.

[0051] Die Ergebnisse des Hardware-Blocks 200 zur SINR-Schätzung sowie des Blocks 250 für die Kanalzustands-schätzung werden abgesehen vom MRC auch im DSP (Firmware-Bereich) zur Weiterverarbeitung benötigt. Im Firmware-Bereich wird beispielsweise die in der Hardware ermittelte Signalleistung durch die Rauschleistung dividiert und daraus der dekadische Logarithmus gebildet, woraus sich ein zellspezifischer Schätzwert für das Signal-zu-Rausch-und-Interferenz-Leistungsverhältnis SINR in Dezibel (dB) ergibt.

[0052] Es wird ferner darauf hingewiesen, dass die in den Gleichungen (1) bis (5) angegebenen Signalverarbeitungs-Algorithmen exemplarisch sind, da sie nur für einen bestimmten Betriebsmodus (normal mode, data based) gelten und in anderen Betriebsmodi unterschiedlich sein können.

[0053] Aus dem Vorhergehenden wird deutlich, dass die Aufgabe der in Fig. 3 dargestellten Hardware-Blöcke die Vorverarbeitung der Eingangssignale durch relativ einfache mathematische Operationen, die aber zu einer erheblichen Datenreduktion führen, ist. Dadurch wird erreicht, dass für die Weiterverarbeitung der Ergebnisdaten im DSP nur wenige, dafür aber komplexere mathematische Operationen ausgeführt werden müssen. Diese können leicht in Firmware programmiert werden.

[0054] Gemäß der Erfindung wurde erkannt, dass es ineffizient wäre, die "idealen" Algorithmen nach den Gleichungen (1) bis (5) direkt in Hardware umzusetzen, weil für die Skalierungsoperationen Divisionen durch Zahlen benötigt werden, die keine Zweierpotenzen darstellen. Divisionen durch 10 oder durch eine variable Anzahl von Symbolen oder Rake-Fingern, wie sie in den Gleichungen (1) bis (5) vorkommen, sind in Firmware wesentlich leichter durchzuführen als in Hardware. Daher werden in der Hardware nur Divisionen durch Zweierpotenzen, d.h. Schieberegisteroperationen, durchgeführt. Divisionen durch beliebige positive ganze Zahlen, Z , werden durch Divisionen durch die "nächstgelegene" Zweierpotenz $2^{\text{round}(\log_2(Z))}$ ersetzt. Statt durch 10 zu dividieren, wird also um drei Bits nach rechts geschoben, was einer Division durch 8 entspricht. Die Ergebnisse $W_D(C)$, $A(C)$, $N(C)$ werden fehlerbehaftet an den DSP (Firmware-Bereich) weitergeleitet. Der Approximationsfehler wird in der Firmware (durch den DSP) korrigiert, wo Divisionen durch beliebige Zahlen viel einfacher als in Hardware durchführbar sind. Der Korrekturfaktor, mit dem in der Firmware die Ergebnisse aus der Hardware multipliziert werden müssen, lautet $2^{\text{round}(\log_2(Z))}/Z$, ist also gleich dem Quotienten aus approximiertem und tatsächlichem Nennerausdruck.

[0055] Insofern sind die Skalierer $sc12$, $sc22$, $sc33$ im Wesentlichen Barrel-Shifter, also Hardware-Einheiten, die eine programmierbare Anzahl von Bits nach rechts (oder links) schieben können und so die Skalierungsfaktoren in den Gleichungen (1) bis (3) approximieren. Die Anzahl der Bits, um die geschoben werden muss, entspricht der in Fig. 2 dargestellten Größe sv (Stellenverschiebung).

[0056] Neben den Schiebeoperationen dienen die Skalierer $sc12$, $sc22$, $sc33$ auch dazu, das Signal bei Erreichen definierter Schwellwerte zu sättigen (Saturation), um Signal-Überläufe zu verhindern und um von der Genauigkeit her nicht erforderliche LSBs durch Quantisieren zu unterdrücken.

[0057] Im Gegensatz zu den Skalierern $sc12$, $sc22$, $sc33$ ist der Skalierer cse_scale in dem Block 250 für die Kanalzustandsschätzung in konventioneller Weise, d.h. als Hardware-Multiplizierer gemäß Fig. 1, ausgeführt. Der Grund hierfür besteht darin, dass der Normierungsfaktor $F(C)$ direkt in der Hardware benötigt wird und eine Division in Firmware hier zu viel Zeit benötigen würde (Schleife Hardware-Firmware-Hardware). Daher wird in der Firmware der Korrekturfaktor berechnet, der sich aus dem Produkt aller Approximationsfehler der Skalierungsfaktoren in den Skalierern $sc12$, $sc22$, $sc33$ ergibt. Dieser Wert ist im Allgemeinen keine Zweierpotenz. Deshalb ist beim Skalierer cse_scale ein Hardware-Multiplizierer erforderlich, um das von dem Dividierer DIV ausgegebene Datensignal mit einem vom DSP berechneten und in ein Register (nicht dargestellt) programmierten Skalierungsfaktor zu multiplizieren.

[0058] Sowohl bei der Schaltung in Fig. 2 als auch bei der Schaltung in Fig. 3 wird der Korrekturfaktor K_f also in Firmware berechnet. Der Unterschied zwischen den beiden Schaltungsanordnungen besteht darin, dass in Fig. 2 der Korrekturblock 100 in Firmware ausgeführt ist, während in Fig. 3 für eine der berechneten Größen, $F(C)$, der Korrekturblock durch den Skalierer cse_scale repräsentiert wird, d.h. in Hardware realisiert ist.

[0059] Zum besseren Verständnis der Erfindung wird diese an einem einfachen Zahlenbeispiel verdeutlicht: Es wird das DPCH-Zeitschlitzformat (slot format) 15 angenommen. Laut dem eingangs angegebenen UMTS-Standard 3GPP TS 25.211 (2002-09) gibt es in diesem Fall $K_x = 304$ Datensymbole im Zeitschlitz. Es wird eine Zelle mit 7 Rake-Fingern zu demodulierenden Ausbreitungspfaden zugrunde gelegt, d.h. $M(C) = 7$. Um den Skalierer $sc12$ zu programmieren, berechnet man die nächstliegende Zweierpotenz zu 304, also $2^{\text{round}(\log_2(304))} = 2^{\text{round}(8,2479...)} = 2^8 = 256$. Daher wird im

Skalierer sc12 mit dem Faktor $1/256$ multipliziert. Zur Ansteuerung des Skalierers sc12 wird lediglich die Anzahl der Bit-Verschiebungen, also die Zahl 8, in das Skalierungsparameterregister (entspricht dem Register 70 in Fig. 2) programmiert. Dies erfordert im Vergleich zu der Zahl 256 eine deutlich kleinere Bitbreite für die Zahlendarstellung, d.h. $B_i \ll B_k$ in Fig. 2.

[0060] Der Approximationsfehler in dem Subblock 210 für die DPCH-Leistungsberechnung ist folglich ein Faktor $304/256$, d.h. die Ergebnisse sind um diesen Faktor zu groß.

[0061] Der Approximationsfehler in dem Subblock 220 für die CPICH-Leistungsberechnung ist ein Faktor $10/8$, da in dem Skalierer sc33 statt durch 10 nur durch 8 dividiert wird.

[0062] Der Approximationsfehler in dem Subblock 230 für die CPICH-Rauschleistungsberechnung mit dem Skalierer sc22 ergibt einen Faktor $70/64$ (da $10 \cdot M(C) = 10.7 = 70$ durch 64 ersetzt wird). Daraus ergeben sich die folgenden Korrekturfaktoren:

Korrekturfaktor für die Signalleistung (Subblock 210):

$$Kf_1 = 256/304 = 0,84210...$$

Korrekturfaktor für die Rauschleistung (Subblock 230):

$$Kf_2 = 64/70 = 0,91428...$$

Korrekturfaktor für das Amplitudenverhältnis (Subblock 240):

$$Kf_3 = \sqrt{(256/304)/(8/10)} = 1,02597...$$

Daher ist für die Berechnung des Normierungsfaktors in dem Block 250 ein Korrekturfaktor

$$Kf_4 = \sqrt{((256/304)/(8/10))/(64/70)} = 1,12216... \text{ zu verwenden.}$$

[0063] Die Multiplikationen mit den ersten drei Korrekturfaktoren Kf_1 , Kf_2 , Kf_3 bezüglich der Ausgabegrößen $W_D(C)$, $N(C)$, $A(C)$ werden rein in Firmware durchgeführt (mittels des DSPs, der dem Software-Korrekturblock 100 in Fig. 2 entspricht), die Multiplikation mit dem letzten Korrekturfaktor Kf_4 erfolgt, wie bereits erwähnt, in Hardware durch den Skalierer cse_scale. Der Korrekturfaktor Kf_4 als solcher wird aber in Firmware berechnet und über ein Register (nicht dargestellt) in dem Hardware-Block 250 als Parameter programmiert.

Patentansprüche

1. Verfahren zur Durchführung einer Multiplikations- oder Divisionsoperation $X \cdot K$ bzw. $X \cdot 1/K$ in einer elektronischen Schaltung, welche einen Software-Schaltungsbereich und einen Hardware-Schaltungsbereich umfasst, wobei X und K reelle Zahlen in einer Zahlendarstellung zur Basis p , insbesondere $p = 2$, sind, mit den Schritten:

- Berechnen einer Stellenverschiebung sv in dem Software-Schaltungsbereich (50), derart, dass p^{sv} ein Näherungswert für K ist;
- Verschieben des Wertes X bei einer Multiplikation um sv Stellen nach links bzw. bei einer Division um sv Stellen nach rechts in dem Hardware-Schaltungsbereich (80);
- Berechnen eines geeigneten Korrekturfaktors Kf in dem Software-Schaltungsbereich (50) der elektronischen Schaltung; und
- Multiplizieren (100) des verschobenen Wertes von X mit dem Korrekturfaktor Kf .

2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet, dass** die Stellenverschiebung $sv = \text{round}(\log_p K)$ ist.

3. Verfahren nach Anspruch 1 oder 2, **dadurch gekennzeichnet, dass** der Korrekturfaktor Kf bei einer Division den Wert $(p^{sv})/K$ oder bei einer Multiplikation den Wert K/p^{sv} aufweist.

4. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet, dass** der Wert X sich zeitlich wesentlich schneller ändert als der Wert K .

5. Verfahren nach einem der vorhergehenden Ansprüche,

dadurch gekennzeichnet, dass

das Multiplizieren des verschobenen Wertes von X mit dem Korrekturfaktor Kf im Software-Schaltungsbereich (100) erfolgt.

6. Verfahren nach einem der vorhergehenden Ansprüche,
gekennzeichnet durch den weiteren Schritt: Sättigen des verschobenen Wertes von X, sofern dieser einen vordefinierten Zahlenwert erreicht hat.

7. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet, dass
das Verschieben des Wertes X mittels eines Barrel-Shifter (80) durchgeführt wird.

8. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet, dass
das Verfahren zur Berechnung der Leistung eines Signals in einem Mobilfunknetz, insbesondere des über den DPCCH-Kanal übertragenen Signals und/oder des über den CPICH-Kanal übertragenen Signals in UMTS, vorgesehen ist.

9. Elektronische Schaltung zur Durchführung einer Multiplikations- oder Divisionsoperation $X \cdot K$ bzw. $X \cdot 1/K$, welche einen Software-Schaltungsbereich und einen Hardware-Schaltungsbereich umfasst, wobei X und K reelle Zahlen in einer Zahlendarstellung zur Basis p, insbesondere $p = 2$, sind, mit

- ersten Berechnungsmitteln (50) des Software-Schaltungsbereichs zum Berechnen einer Stellenverschiebung sv in dem Software-Schaltungsbereich, derart, dass p^{sv} ein Näherungswert für K ist,
- einem in dem Hardware-Schaltungsbereich enthaltenen steuerbaren Stellenverschieber (80) zum Verschieben des Wertes X bei einer Multiplikation um sv Stellen nach links bzw. bei einer Division um sv Stellen nach rechts,
- zweiten Berechnungsmitteln (50) des Software-Schaltungsbereichs zum Berechnen eines geeigneten Korrekturfaktors Kf, und
- Mitteln (100) zum Multiplizieren des verschobenen Wertes von X mit dem Korrekturfaktor Kf.

10. Elektronische Schaltung nach Anspruch 9;
dadurch gekennzeichnet, dass
der Wert $sv = \text{round}(\log_p K)$ ist.

11. Elektronische Schaltung nach Anspruch 9 oder 10,
dadurch gekennzeichnet, dass
der Korrekturfaktor Kf bei einer Division den Wert $(p^{sv})/K$ oder bei einer Multiplikation den Wert K/p^{sv} aufweist.

12. Elektronische Schaltung nach einem der Ansprüche 9 bis 11,
dadurch gekennzeichnet, dass
der Wert X sich zeitlich wesentlich schneller ändert als der Wert K.

13. Elektronische Schaltung nach einem der Ansprüche 9 bis 12,
dadurch gekennzeichnet, dass
die Mittel (100) zum Multiplizieren des verschobenen Wertes von X mit dem Korrekturfaktor Kf im Software-Schaltungsbereich realisiert sind.

14. Elektronische Schaltung nach einem der Ansprüche 9 bis 13,
gekennzeichnet durch
eine Sättigungsstufe zum Sättigen des verschobenen Wertes von X.

15. Elektronische Schaltung nach einem der Ansprüche 9 bis 14,
dadurch gekennzeichnet, dass
der steuerbare Stellenverschieber ein Barrel-Shifter (80) ist.

16. Elektronische Schaltung nach einem der Ansprüche 9 bis 15,
dadurch gekennzeichnet, dass
die Schaltung zur Berechnung der Leistung eines Signals in einem Mobilfunknetz, insbesondere des über den

DPCH-Kanal übertragenen Signals und/oder des über den CPICH-Kanal übertragenen Signals in UMTS, ausgelegt ist.

17. Elektronische Schaltung zur Durchführung einer SINR-Schätzung in einem UMTS-Kommunikationssystem, mit

- einem ersten Hardware-Block (210) zur Durchführung einer DPCH-Leistungsberechnung,
- einem zweiten Hardware-Block (220) zur Durchführung einer CPICH-Leistungsberechnung,
- einem dritten Hardware-Block (230) zur Durchführung einer Rauschleistungsberechnung, und
- einem Prozessor (DSP) zur Weiterverarbeitung von Daten basierend auf den in den Hardware-Blöcken (210, 220, 230) berechneten Größen, wobei die ersten, zweiten und dritten Hardware-Blöcke (210, 220, 230) Hardware-Schaltungsbereiche und der Prozessor Software-Schaltungsbereiche gemäß einem der Ansprüche 9 bis 16 realisieren.

FIG 1

Stand der Technik

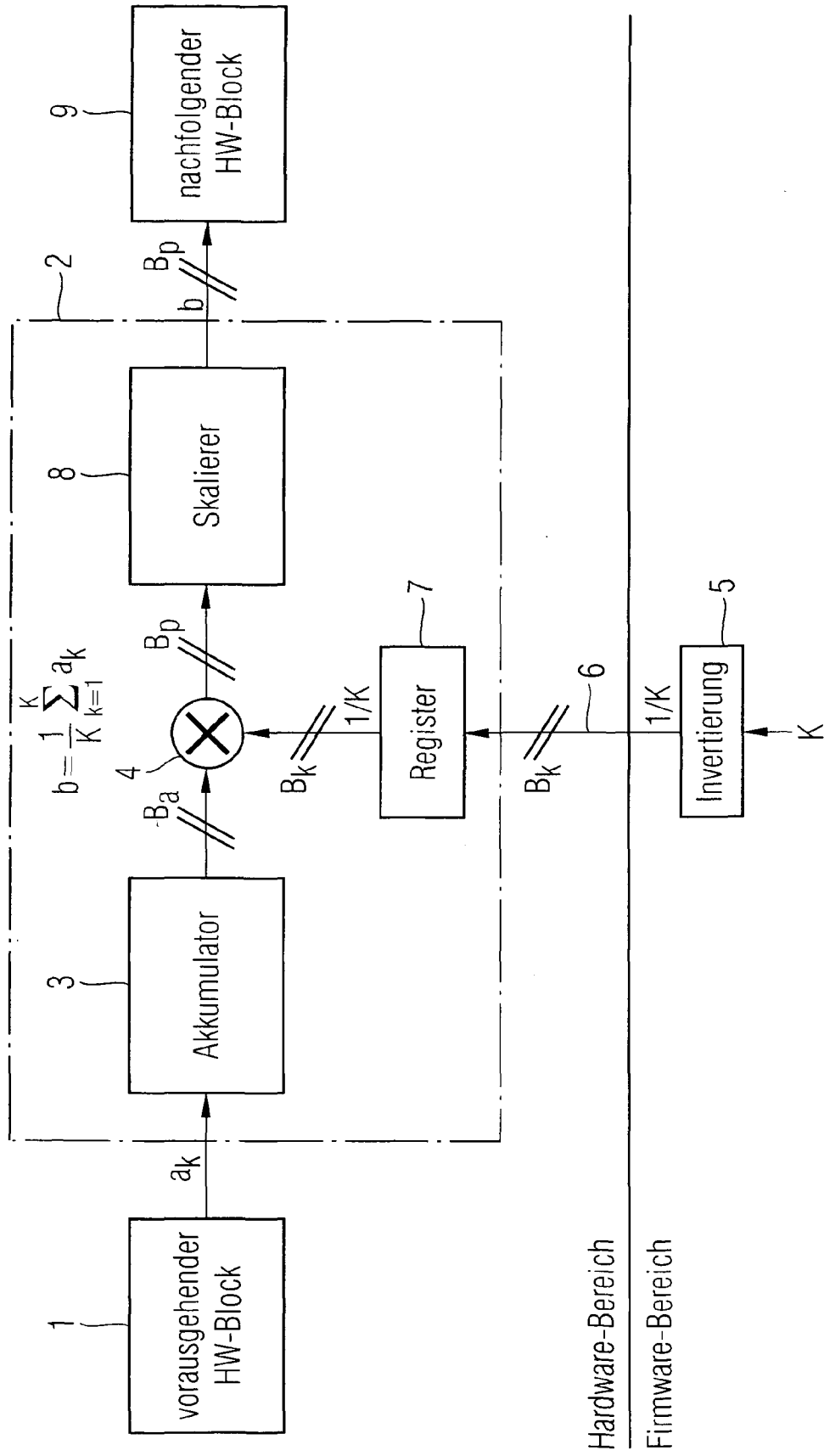


FIG 2

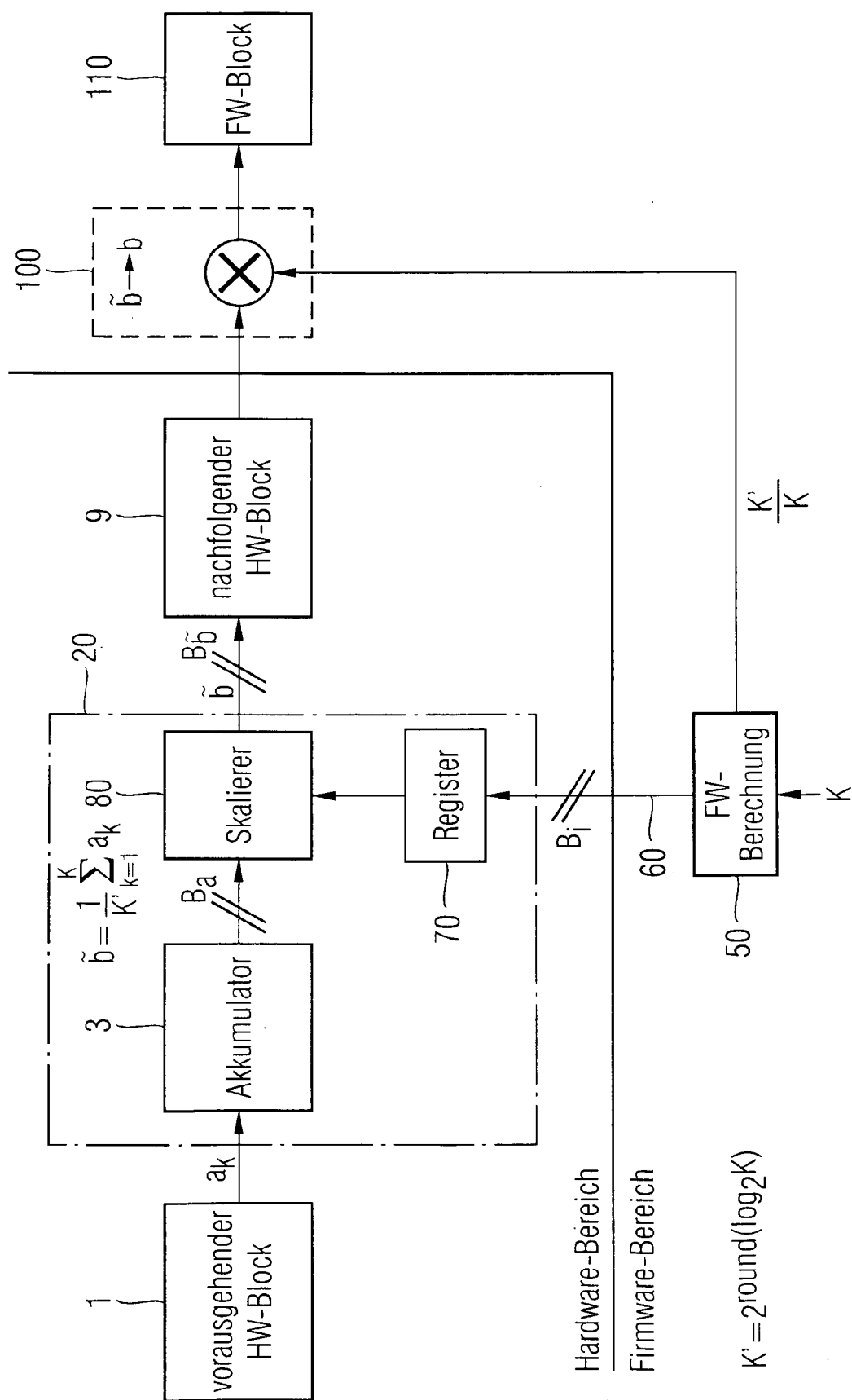
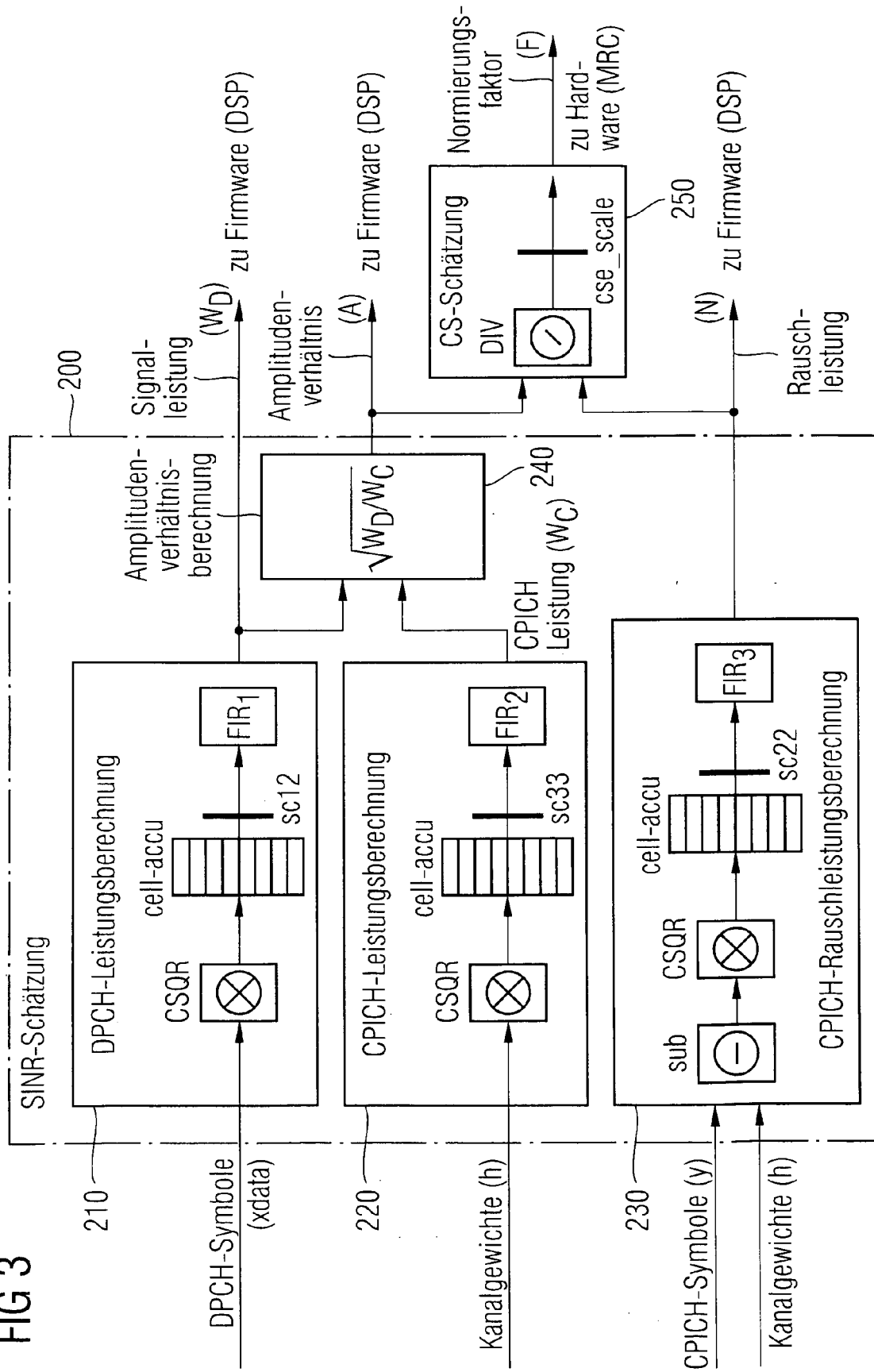


FIG 3





Europäisches
Patentamt

EUROPÄISCHER RECHERCHENBERICHT

Nummer der Anmeldung
EP 05 02 5055

EINSCHLÄGIGE DOKUMENTE			
Kategorie	Kennzeichnung des Dokuments mit Angabe, soweit erforderlich, der maßgeblichen Teile	Betrifft Anspruch	KLASSIFIKATION DER ANMELDUNG (IPC)
X	US 2003/131037 A1 (FREIDHOF MARKUS) 10. Juli 2003 (2003-07-10) * Zusammenfassung * * Abbildungen 1-3 * * Absatz [0020] * * Absatz [0022] *	1-17	G06F7/52
X	US 2004/186871 A1 (ETTORRE DONATO ET AL) 23. September 2004 (2004-09-23) * Zusammenfassung * * Abbildungen 1-3 * * Absatz [0026] - Absatz [0051] *	1-17	
A	US 4 507 676 A (DISCHERT ET AL) 26. März 1985 (1985-03-26) * Zusammenfassung * * Abbildungen 1,2 * * Spalte 1, Zeile 50 - Spalte 2, Zeile 33 *	1-17	
			RECHERCHIERTE SACHGEBIETE (IPC)
			G06F
Der vorliegende Recherchenbericht wurde für alle Patentansprüche erstellt			
Recherchenort München		Abschlußdatum der Recherche 23. Dezember 2005	Prüfer Pfab, S
KATEGORIE DER GENANNTEN DOKUMENTE X : von besonderer Bedeutung allein betrachtet Y : von besonderer Bedeutung in Verbindung mit einer anderen Veröffentlichung derselben Kategorie A : technologischer Hintergrund O : nichtschriftliche Offenbarung P : Zwischenliteratur T : der Erfindung zugrunde liegende Theorien oder Grundsätze E : älteres Patentdokument, das jedoch erst am oder nach dem Anmeldedatum veröffentlicht worden ist D : in der Anmeldung angeführtes Dokument L : aus anderen Gründen angeführtes Dokument & : Mitglied der gleichen Patentfamilie, übereinstimmendes Dokument			

4
EPO FORM 1503 03.82 (P04C03)

**ANHANG ZUM EUROPÄISCHEN RECHERCHENBERICHT
 ÜBER DIE EUROPÄISCHE PATENTANMELDUNG NR.**

EP 05 02 5055

In diesem Anhang sind die Mitglieder der Patentfamilien der im obengenannten europäischen Recherchenbericht angeführten Patendokumente angegeben.
 Die Angaben über die Familienmitglieder entsprechen dem Stand der Datei des Europäischen Patentamts am
 Diese Angaben dienen nur zur Unterrichtung und erfolgen ohne Gewähr.

23-12-2005

Im Recherchenbericht angeführtes Patendokument	Datum der Veröffentlichung	Mitglied(er) der Patentfamilie	Datum der Veröffentlichung
US 2003131037 A1	10-07-2003	DE 10163350 A1	17-07-2003
		JP 2003263423 A	19-09-2003

US 2004186871 A1	23-09-2004	CA 2457199 A1	27-02-2003
		CN 1545652 A	10-11-2004
		EP 1417564 A2	12-05-2004
		WO 03017084 A2	27-02-2003
		IT T020010817 A1	17-02-2003
		JP 2005500613 T	06-01-2005

US 4507676 A	26-03-1985	AU 560272 B2	02-04-1987
		AU 2080283 A	03-05-1984
		CA 1191960 A1	13-08-1985
		DE 3339029 A1	03-05-1984
		ES 8502303 A1	16-03-1985
		FR 2535567 A1	04-05-1984
		GB 2131579 A	20-06-1984
		IT 1171789 B	10-06-1987
		JP 59108146 A	22-06-1984

Für nähere Einzelheiten zu diesem Anhang : siehe Amtsblatt des Europäischen Patentamts, Nr.12/82